Rainfall Prediction Using LSTM Deep Learning Model

Tuan Nguyen
George Mason University
Fairfax, VA 22030
tnguy209@gmu.edu

Abstract

In order to bridge the gap between research on short-term rainfall predictions (up to 14 days) and long-term predictions without compromising temporal information of precipitation events, this paper proposed using LSTM models to predict day-by-day rainfall amount in millimeters up to 30 days future period using data gathered in period from 2007 - 2017 from 21 stations across Australia. Two models' variants were created, the Daily Iterative model which use prior sequence to predict 1 day ahead, which can be used iteratively to predict long-term rainfalls, and the Single Prediction model where the model output 30 days future rainfall sequence at once. Comparison with 0-predictors and random weighted model showed that both model variants were able to capture some patterns between historical record data and future rainfall amount. The Daily Iterative model tends to overpredict rainfall amount but underpredict the occurrence of rainfall events. In contrast, the Single Prediction model makes a lot of small rainfall predictions, even during days which no rainfall events occurred. Both models exhibit limitations which would be possible to address using more advanced model architecture such as modified LSTM or Transformer.

1 Introduction

Rainfall, or precipitations, are important in many different aspects of life and can have impacts in many industries such as agriculture, environment, industrial productions, constructions, transportation commercial airlines, ...etc. Excess or lack of rainfall can cause short and long term problems to affected areas in the form of floods or draughts. Therefore, accurately predicting in advance when rainfall events would occur can help mitigate economic and infrastructure damage as well as avoid human life lost.

The most common method of rainfall prediction currently employed globally is based on physical simulation models (more commonly known as the field of meteorology), where other weather information (cloud radar images, water vapor satellite images, ocean currents, temperature, windspeed, …etc.) is used to construct a simulation of the cloud movement and atmospheric conditions, which will then use to predict the probability of rainfall occurring at a given time interval.

A new emerging trend is to use statistical models to directly predict rainfall without relying on simulation output. These types of models use historical recorded weather data or utilize satellite and radar data as input to predict future weather conditions. In the case of rainfall prediction, different ML/DL models such as SVM [3], MLP [4], ANFIS [6], LSTM [2] [5] have been used to predict rainfall with meteorological parameters such as windspeed, humidity, dew point, temperature as input. Others have employed CNN [7], ConvLSTM [8] or their combination [9], on satellite and radar images to perform short term rainfall forecasting. Existing literature in the past 5 years has shown that LSTM architecture has proven to be effective in modeling relationships between current hydrological measurements and future rainfall events.

Several mentioned papers [1] [2] have achieved good performance in predicting short term rainfall predictions (up to 14 days ahead, definition varies depending on agency). However, for longer periods (more than 14 days), the model performance is largely dependent on the quality of recorded data, the location selected to perform prediction as well as its climate characteristics. The main difficulty in long-term rainfall predictions is caused by the "butterfly effect" that exist in weather conditions, where a small changes in initial conditions can lead to significant changes in future conditions, which in turn makes long term forecast volatile and less predictable. While some papers attempted to decrease this volatility by combining daily records into monthly or weekly average and had achieved good prediction result with monthly average rainfall, the temporal significant of rainfall event are lost in this aggregation, which can be detrimental to precipitation sensitive activities in agriculture or disaster management.

This work proposes a possible solution to bridge this gap between short term daily precipitation forecast and long term monthly average rainfall prediction by attempting to create models that would predict future rainfall for a region in millimeters (daily iterative prediction model and single prediction model) using data from 30 - 90 days prior to the interested period. These models are based on LSTM network architecture, which has shown promising result for long term time series forecasting in general. Results from the models would then be compared to all-0 prediction, randomized weight model, and similar models from reviewed literatures.

This paper will begin by describing the dataset and performed preprocessing methods (Section 2), before going into details of model architecture and training process (Section 3), which is followed by comparison between model's result, all-zero predictors, and randomized weight model (section 4). The final section of the paper contains discussion on model behavior and potential next step for this project (section 5).

2 Data set and preprocessing

The main dataset used in this project is part of the "Rain in Australia" [10] from Kaggle, which itself is drawn from numerous weather stations across Australia and from the Australian Bureau of Meteorology. The full dataset comprises of 145,460 observations gathered over 10 years period (2007 – 2017) from 49 different weather stations across the country, contains daily measurements for: Temperature (max, min, 9am, 3pm), Rainfall, Evaporation, Sunshine hours, Wind gust speed & directions (9am, 3pm), Humidity (9am, 3pm), Pressure (9am, 3pm), Cloud (9am, 3pm).

The dataset contains a high number of missing values, with several stations lacking all data for evaporation and sunshine hours (Appendix A). All stations with more than 30% missing values in a column were removed from the dataset, with the exception of cities with large population such as Sydney and Melbourne. The remaining 21 stations are split into separate data frames for each station, in which missing values in each column of each station are filled using respective column multiyear modes and means. This ensures local uniformity as the multiyear mean and mode of each measurement is different in each location. The completed data frames for each station are joined into a large data frame of 64,353 observations, which served as the primary input for the model.

Figure 1: Selected stations location within Australia

According to NOAA rainfall intensity classification for 24 hours period [11], any day that has a total precipitation exceeding 150mm is considered "Extremely Heavy Rain", which can be considered as an extreme outlier. Out of 64,353 observations, there are only 25 days which has a rainfall record exceed 150mm. Thus, to ensure model performance is not negatively affected by these extreme outliers, their observation's Rainfall measurements are fixed to exactly 150mm.

The dataset is then rescaled using Sklearn's MinMaxScalar for Rainfall, and StandardScalar for remaining attributes except for Humidity, which was converted from percentage to ratio between 0 and 1.

The scaled data set was then transformed into smaller fixed length time series by using sliding window to create smaller sequence with length of 90 days with 60 days input and 30 days prediction (or sequence with length of 61 for daily iterative model). New sequences are created by sliding the window 1 day forward each time. The entire dataset was transformed into a list of sequences, which were converted to tensors of size [data length × 17 (16 measurements + month) × sequence length] suitable for model input.

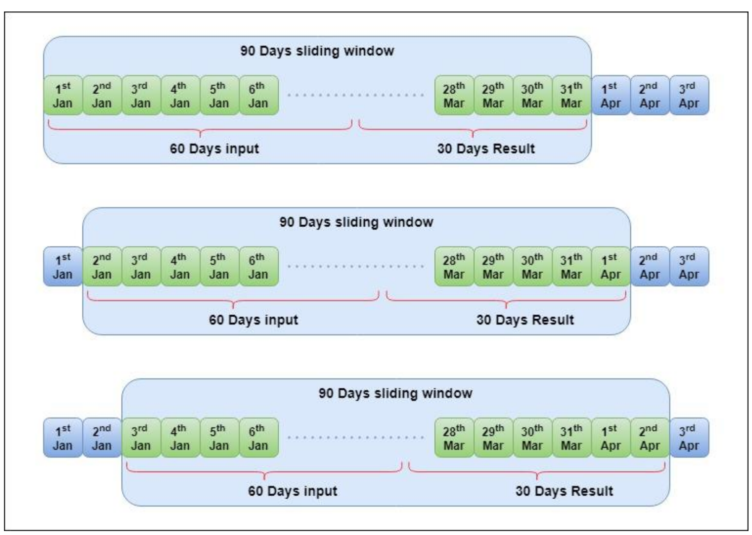

Figure 2: Example of creating sequence from sliding window

After this transformation, the dataset was split into training, validation, and testing set, with respective ratios of 0.8, 0.15, 0.05 each. The final training, validation, and testing set contains 56,232, 10,544 and 3,514 sequences respectively (for 60 days). Finally, Torch Data Loader was used to pre-load and store data during model training, validation, and testing.

3 Model

3.1 Model Architecture

LSTM (Long Short-Term Memory) is a recurrent neural network architecture that has emerged as a powerful tool for modeling long-term dependencies in sequence data. The main difference between LSTM and RNNs is the introduction of memory cell and gating mechanism that allows LSTM to retain information over longer sequences. The memory cell is the core component of the LSTM, storing information over extended periods of time in the form of long-term and short-term memories, while the gates regulate the flow of information into and out of the cell. This architecture allows LSTMs to capture complex dependencies in data, making them particularly well-suited for applications such as natural language processing and time series analysis.

The LSTM forms the core of this model, as its ability to retain information over time played a vital role in identifying patterns and relationship between meteorological measurements and future rainfall events. The model was created and trained using "PyTorch" Python library. The initial version of the rainfall predictor contains a single LSTM layer followed by 2 fully connected layers, with this model only using numerical measurements as input for each time step. The current version of the model contains an additional fully connected layer, which acted as a simple embedding for the categorical input "Month". This layer output would be added back to the last hidden state of the LSTM layer before it is used as final input for 2 fully connected layers to get final prediction result. ReLU activation function was used for all fully connected layers and LSTM hidden layer output.

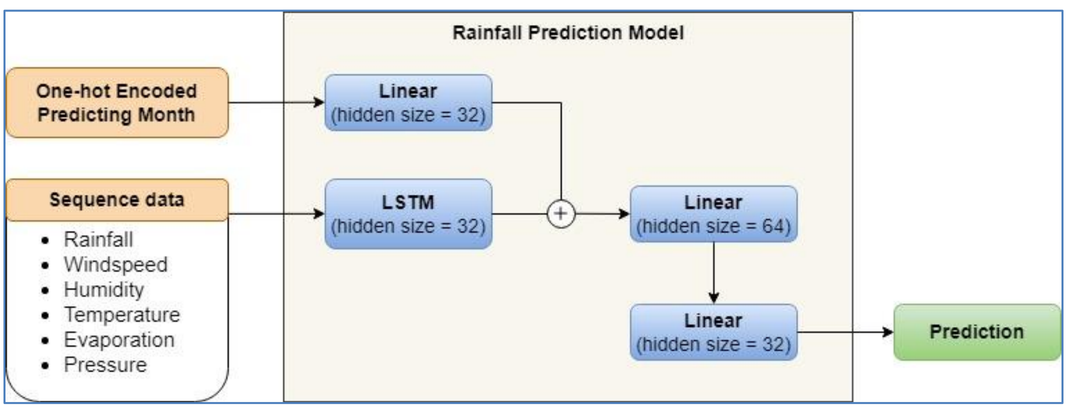

Figure 3: Rainfall Prediction Model Workflow

3.2 Loss function

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

The 2 most widely known loss function for regression problems are Mean Squared Error (MSE) and Mean Absolute Error (MAE). MSE measures the average squared difference between the predicted values and the actual values, while MAE measures the average absolute difference between them. In general, MSE tends to penalize loss from outlier in predictions and incentivizes the model to make predictions that are closer to the mean. In contrast, MAE is more robust to outliers and is often favored when the focus is on predicting the median of the distribution rather than the mean, while not punishing outlier prediction as much compared to MSE. Since rainfall events are considered a form of outlier due to their frequency of appearance, and due to the characteristic of rainfall data, which are often zero-inflated, MSE loss is preferred, however, it is equally important to accurately predict the magnitude of significant rainfall events as these events often have higher degree influence on related sectors

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{for} |y - f(x)| \le \delta, \\ \delta\,|y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$$

The Huber Loss combines the advantages of both MSE and MAE by balancing both loss function. A threshold, delta (normally set to 1), is selected, where if the absolute difference between the target value and prediction is less than delta, the final loss would be calculated using MSE function, and vice versa. The Huber Loss effectively magnify losses that are greater than delta, focus more on samples with higher loss than delta, while paying less attention to samples with losses lower than delta. Using Huber loss allows the model to retain the ability to predict outliers, while not overestimating other non-outlier samples.

3.3 Model Training

3.3.1 Daily Iterative Model

The daily iterative model only predicts 1 future day at a time. The models were trained for 50 epochs with Adam optimizer default parameters, a learning rate of 0.001, and the best weight parameters with lowest validation loss were recorded and used as final weight for each model. Various batch sizes and input sequence length were tested to measure model's performance on test set.

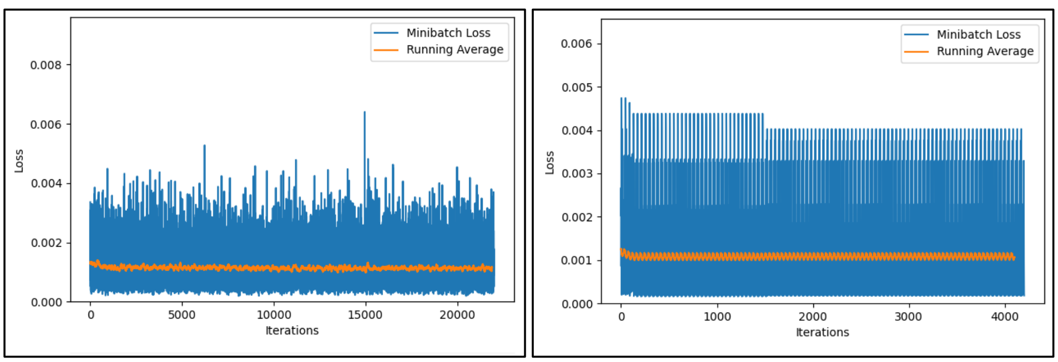

Figure 4: Training loss (left) and Validation loss (right) for Daily Iterative Model at batch size = 256, input length = 60

3.3.2 Single Prediction Model

The single prediction model output result of 30 days prediction at once. This was done by modifying the last fully connected layer to have output size = 30. The model was trained for 50 epochs with Adam optimizer, a learning rate of 0.001 and batch size of 128.

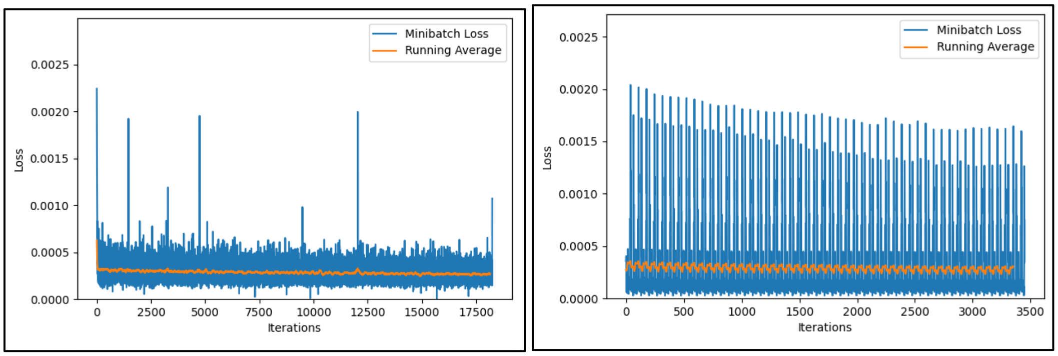

Figure 5: Training loss (left) and Validation loss (right) for Single Prediction Model at batch size = 128, input length = 60

4 Experimental Results

4.1 Study Area

The study area comprises of 21 stations from Australia, with majority of the stations located in coastal region of southern Australia: Sydney Airport, Sydney, Wagga Wagga, Moree, Cobar, Melbourne Airport, Mildura, Portland, Watsonia, Dartmoor, Brisbane, Cairns, Townsville, Nuriootpa, Perth Airport, Perth, Alice Springs, Darwin, Norfolk Island, Mount Gambier, Hobart.

This region is characterized by a temperate climate with rainfall throughout the year. Southern Australia receives its rainfall primarily from winter frontal systems that move across the Southern Ocean, with occasional summer rainfall from convective thunderstorms. Precipitation in Southern Australia is generally higher along the coastal areas, which are affected by westerly winds and presence of high mountains that force the air to rise and cool, resulting in rainfall events. The inland areas, on the other hand, are much drier, with some regions receiving less than 250 mm of rainfall annually. The variability of rainfall across the region can lead to significant challenges in precipitation prediction.

4.2 Evaluation

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

The root mean square error (RMSE) is commonly used to reflect the total error of prediction results. MSE and MAE can also be used to compare performance between different configurations of the model. The below table shows the performance of Daily Iterative Model for various training configurations, and prediction result for the test set.

4.2.1 Daily Iterative Model

Table 1: Daily Iterative Model Performance for different Batch size and Sequence Length

Batch Size	Metric	Input Sequence Length		
		30 days	60 days	90 days
32	MSE	52.81804	56.30755	48.95148
	RMSE	7.26760	7.50383	6.99653
	MAE	2.26170	2.28474	2.00663
64	MSE	61.54731	61.32821	45.00431
	RMSE	7.84520	7.83123	6.70852
	MAE	2.20892	2.25611	2.2923
128	MSE	51.45301	57.52034	46.68670
	RMSE	7.17307	7.58421	6.83276
	MAE	2.30552	2.64371	2.04567
256	MSE	54.02511	57.29244	**45.78643**
	RMSE	7.35017	7.56917	**6.76656**
	MAE	2.49377	2.59183	**2.20926**

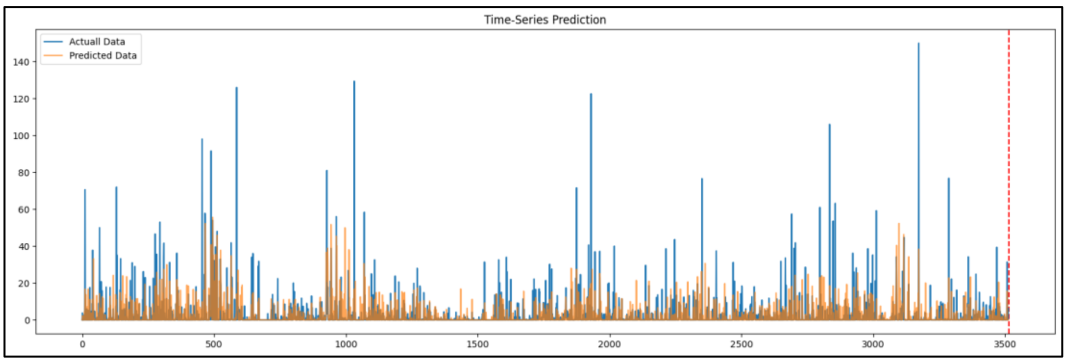

Figure 6: Prediction result from Daily Iterative Model (90days input) on test data

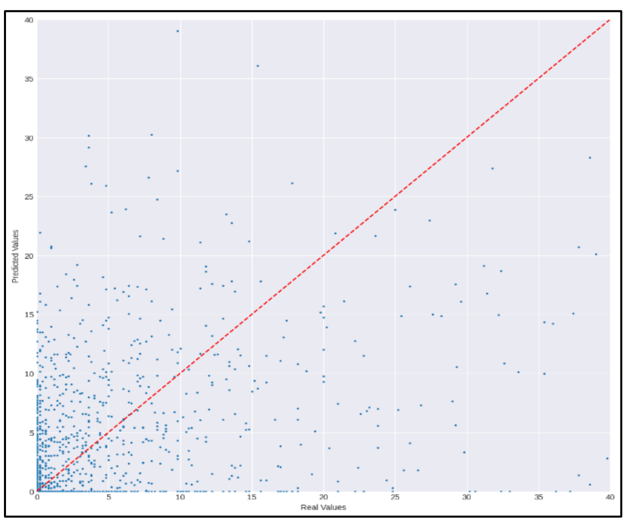

Figure 7 Scatter plot showing Prediction (y-axis) versus Target rainfall (x-axis) for Daily Iterative Model

Using 90 days input sequence length significantly improved model performance compared to 30-days and 60-days input. The model appears to have the ability to capture the general trend in daily precipitation. However, scatter plot of prediction versus real values shows the model tendency to over predicts the magnitude of light rainfall events (<20mm).

4.2.2 Single Prediction Model

The Single Prediction Model was trained with batch size of 128, and achieved MSE = 69.4754; RMSE = 8.33519 and MAE = 2.9877 on test data.

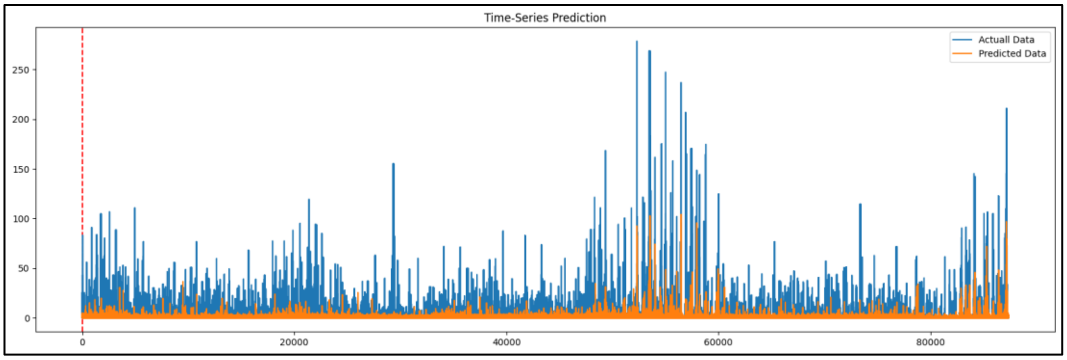

Figure 8: Prediction result from Single Prediction Model on test data

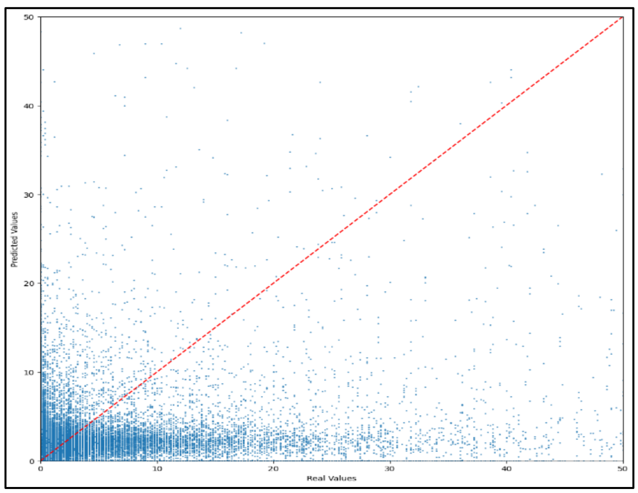

Figure 9: Scatter plot showing Prediction (y-axis) versus Target rainfall (x-axis)

Figure 9 shows the scatter plot between prediction and target values. While the model seems to have captured some patterns within the training data, the majority of predictions lie under the diagonal line, indicating that the model severely under predicts rainfall values. The single prediction model tends to predict small rainfall events for many days. This could be a result of the model attempting to minimize loss by spreading out errors instead of concentrating on predicting large rainfall events which could have resulted in larger loss.

4.2.2 Single Prediction Model

A method to test if the model has captured any relationship between the predictors and the target is to measure model's performance against a predictor that only predicts 0, and randomized weights model.

Table 2: Performance metric comparison between Trained model, 0-predictor, and randomized weights model

Model (best)	Metric	Model Type	
		Daily Iterative	Single Prediction
Trained Model	MSE	45.78643	69.4754
	RMSE	6.76656	8.33519
	MAE	2.20926	2.9877
All zeroes	MSE	75.32471	78.96564
	RMSE	8.67898	8.88626
	MAE	2.37367	2.30722
Randomized	MSE	137.27216	392.0345
	RMSE	11.71632	19.79986
	MAE	9.8348	11.92328

Trained models' performance is better compared to the other 2 predictors, indicating that the trained models have the ability to imitate the relationship between past meteorological measurements and future rainfall to a certain degree. While the trained Daily Iterative model outperformed the other 2 variants in all aspects, the trained Single predictor has worse MAE, but higher RMSE compared to 0-predictor. This behavior is in line with rainfall data and loss function characteristics. Rainfall data are often zero inflated and precipitation are uncommon events, thus, the Huber loss attempt to balance between optimizing MSE and MAE, by sacrificing MAE (making non-zeroes predicts) to gain MSE (predict closer to real value). As a result, the Single Prediction model made a lot of small non-zeroes predictions for its 30 days output sequence.

5 Discussion

Two variants of the same model architecture were created for predicting 1-day ahead (Daily Iterative) and 30-days ahead (Single Prediction). This section will discuss findings related to model training, behavior, limitations and propose possible future work.

5.1 Model Training and behavior

In addition to the Huber loss, MSE loss and MAE loss were also tested during model training. MSE loss tends to steer model aways from making predictions that include zeroes, while MAE loss does the opposite and guide models produce only zeroes predictions. Another aspect of model training that is not often considered is the sample order. When the shuffle seed for training data was not set, the model tends to behave unpredictably and would fall into local optima. From this symptom, it is inferred that certain samples within the training set have significant impact on the model behavior depending on the order of their appearance.

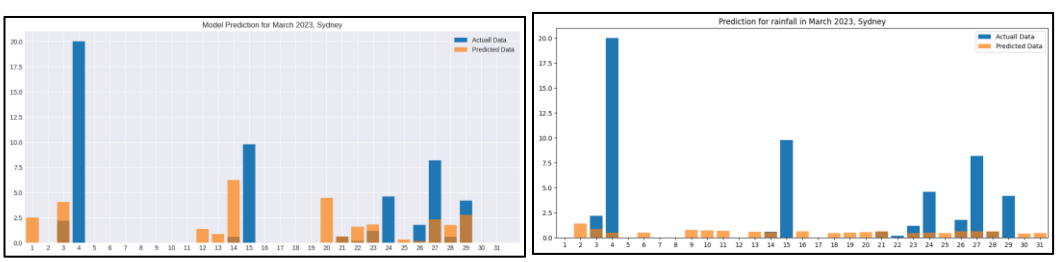

Figure 10: Daily Iterative (left) and Single Prediction (right) Model prediction for Sydney, March 2023

Additionally, both models exhibit opposite prediction characteristics. The Daily Iterative model tends to over-predict rainfall amounts. However, it is less likely to predict that it will rain the next day. In contrast, the Single Prediction model tends to predict that there will be a small amount of rainfall frequently but will severely under-predict the actual amount in case precipitation does occur. It is worth noting that the only major difference between the 2 models is the last linear layer output size, which is 1 for Daily Iterative and 30 for Single Prediction. Further testing is required to determine the exact cause of this behavior.

5.2 Limitations

Currently, 21 stations used as input data for the model are selected due to their low percentage of missing values. However, within these selected stations, there are several locations in which their meteorological characteristic might have drastically difference compared to remaining stations, such as Alice Spring, which lies in the interior of Australia, Hobart, which does not lie on mainland Australia, and Norfolk Island, which is a small island located in the Pacific Ocean. These unique locations may not share the same climate patterns as the majority of other selected stations in the Southern Coastal Australia region, and their inclusion may affect the overall predictions of the region.

Another limitation of the current method is related to how the missing values were addressed during the data processing step. Missing data in each location was filled using multiyear mode and mean for period from 2007 to 2017. However, this multiyear mode and mean can have a significant difference compared to monthly or seasonal mode and mean, which was a factor not taken into account during data processing step.

In this project, MinMaxScalar was used to transform rainfall measurements into input with a range between 0 and 1. One issue with using MinMaxScalar to transform rainfall measurements is that rainfall data is often characterized by a highly skewed distribution, with a few extreme values significantly impacting the overall distribution. Linear transformations such as MinMaxScalar do not take into account these unique characteristics of rainfall data and may result in the loss of important information. Therefore, it may be more appropriate to use non-linear transformation such as logarithmic or exponential to better capture the true nature of the rainfall data. Additionally, it may be useful to examine the impact of different scaling techniques on the model's performance and adjust accordingly to achieve the best possible results.

For both model variants, currently, the prediction relies only on the model last hidden stages, which is a suboptimal design approach as important information about the context and history of the input sequence stored in previous hidden stages are discarded and not utilized in the final prediction. Furthermore, the information from early hidden stages might not be retained when the model has processed through the entire sequence length. These limitations are considered common disadvantages of the vanilla LSTM model.

5.3 Future works

In future work, there are several approaches that could be explored to potentially improve the performance of the rainfall prediction model. One approach would be to incorporate attention mechanisms into the existing LSTM architecture. This would allow the model to focus on the most relevant parts of the input sequence, while also allowing input early in the sequence in contributing to the final prediction.

Another possible approach would be to take advantage of the zero-inflated characteristic of the input data. This could be achieved by multiplying the zero-inflated input with the sigmoid unit at each gate in the LSTM cell. This approach could potentially eliminate redundant information being added to the cell state and enable the model to focus more on non-zero input, or rainfall events, within the input sequence.

Alternatively, Transformer-based models could also be explored as suitable candidates for improving the performance of the predictor model. These models have been shown to be effective in capturing long-range dependencies in sequential data and could potentially lead to improved accuracy compared to the current LSTM-based model.

References

[1] F. R. Hashim, N. G. Nik Daud, K. A. Ahmad, J. Adnan, and Z. I. Rizman, "Prediction of rainfall based on weather parameter using artificial neural network," *J. Fundam and Appl Sci.*, vol. 9, no. 3S, p. 493, Jan. 2018, doi: 10.4314/jfas.v9i3s.38.

[2] Z.-Q. Huang, Y.-C. Chen, and C.-Y. Wen, "Real-Time Weather Monitoring and Prediction Using City Buses and Machine Learning," *Sensors*, vol. 20, no. 18, p. 5173, Sep. 2020, doi: 10.3390/s20185173.

[3] Y. Liu, Q. Zhao, W. Yao, X. Ma, Y. Yao, and L. Liu, "Short-term rainfall forecast model based on the improved BP–NN algorithm," *Sci Rep*, vol. 9, no. 1, p. 19751, Dec. 2019, doi: 10.1038/s41598-019-56452-5.

[4] L. Moharana, A. Sahoo, and D. K. Ghose, "Prediction of Rainfall Using Hybrid SVM-HHO Model," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 1084, no. 1, p. 012054, Oct. 2022, doi: 10.1088/1755-1315/1084/1/012054.

[5] S. Poornima and M. Pushpalatha, "Prediction of Rainfall Using Intensified LSTM Based Recurrent Neural Network with Weighted Linear Units," *Atmosphere*, vol. 10, no. 11, p. 668, Oct. 2019, doi: 10.3390/atmos10110668.

[6] F. Mekanik, M. A. Imteaz, and A. Talei, "Seasonal rainfall forecasting by adaptive network-based fuzzy inference system (ANFIS) using large scale climate signals," *Clim Dyn*, vol. 46, no. 9–10, pp. 3097–3111, May 2016, doi: 10.1007/s00382-015-2755-2.

[7] A. P. Barnes, N. McCullen, and T. R. Kjeldsen, "Forecasting seasonal to sub-seasonal rainfall in Great Britain using convolutional-neural networks," *Theor Appl Climatol*, vol. 151, no. 1–2, pp. 421–432, Jan. 2023, doi: 10.1007/s00704-022-04242-x.

[8] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. Woo, "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting".

[9] C. Zhang, H. Wang, J. Zeng, L. Ma, and L. Guan, "Tiny-RainNet : a deep convolutional neural network with bi-directional long short-term memory model for short-term rainfall prediction," *Meteorol Appl*, vol. 27, no. 5, Sep. 2020, doi: 10.1002/met.1956.

[10] Young, J. (2022, May). Rain in Australia, Version 1. Retrieved March 20, 2023 from https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package.

[11] Weather and Climate Services Division, NCHM, "Rainfall intensity classification."